Sterling Integrator

Build Updates

Version 5.1

Sterling Integrator

Build Updates

Version 5.1

Note
Before using this information and the product it supports, read the information in "Notices" on page 37.

Contents

Chapter 1. Introduction to Build Updates

This document provides information about fixes and enhancements provided in IBM® Sterling Integrator Version 5.1. These builds are cumulative and include all fixes and enhancements contained in the previous build.

Chapter 2. Build 5102 or Higher

Create Permissions Enhancement

If you have upgraded from a previous version of the system, the existing permissions are set to other by default. You may need to edit each permission to apply a new permission type.

Before you begin you need to know the following information:

Manufacturer	Device Types Supported
Permission ID	Permission ID for the permission you are creating. Permission ID is the name of the business process, XSLT document, Web template, or resource for which you are setting the permission. Include the extension for the resource after the ID. Required. Permission IDs: • They must be unique. • They are case-sensitive. • The permission ID must match the name of the business process, XSLT document, Web template, or resource. If the permission ID and the name of the resource do not match exactly, you cannot lock down the resource.
Permission Name	Name of the permission you are creating. Required. A permission name does not need to be unique. Permission names are case-sensitive.
Permission Type	Permission type of the permission you are creating. Required. Permission types include: • UI - Allows access to specific menu items in the interface. UI Permissions with a Permission ID prefixed by _DENY_ deny access to that particular resource or action. For example, if you add a permission, _DENY_BPMANAGE to a user or a group, the user or group will not be able to access BP Management UIs • Mailbox - Allows access to specific mailboxes in the application • Template - Allows access to specific Web templates • BP - Allows access to specific business processes • Tracking - Allows access to specific document tracking options • Community - Allows access to specific community management options • Web Service • Service • eInvoicing • Other - Allows access to resources that are not identified by one of the preceding types

To create a permission:

1. From the **Administration** menu, select **Accounts** > **Permissions**.
2. Next to **Create a new Permission**, click **Go!**
3. On the **Permissions** page, enter the **Permission ID**.

4. Enter the **Permission Name**.
5. Select the **Permission Type**.
6. Click **Next**.
7. Review the permission settings.
8. Click **Finish**.

Connect:Direct Troubleshooting Enhancement

Troubleshooting

Sterling Integrator and Connect:Direct are designed to work together, in a seamless and tightly integrated environment. However, as is the case with any application, occasions may occur when you need to troubleshoot certain components or functions.

This section provides general troubleshooting guidelines when configuring and using Sterling Integrator with Connect:Direct.

To troubleshoot general Connect:Direct functions, see the Connect:Direct documentation set for general troubleshooting information and instructions.

You may need to work with your Trading Partners or system administrators to troubleshoot the systems you are communicating with.

Performance at Startup

The Connect:Direct Server adapter (CDSA) sessions can show slowness in execution immediately after startup due to the buildup of database connections. Performance returns to normal after the database connections are pooled in memory. A solution to overcome the initial slowness, is to increase the initial JDBC connection pool sizes.

Verifying Connectivity

To verify connectivity between Connect:Direct and Sterling Integrator:

1. Check configurations on Connect:Direct and Sterling Integrator for compliance. In particular, look at the network maps on both systems.
2. Verify that the Connect:Direct nodes in question are active when a begin session is invoked.
3. Verify that perimeter services client connects to perimeter services server. If the client cannot connect, it causes the perimeter services-enabled Connect:Direct Server adapter to fail. Perimeter servers do not automatically reboot after a failure.

To determine the status of the perimeter services client in Sterling Integrator:

1. From the Sterling Integrator **Operations** menu, select **Troubleshooter**.
2. Look for the Perimeter Servers area. It is usually at the bottom of the **Troubleshooting** page. This shows the state of the perimeter server clients which are defined in Sterling Integrator and whether they are on. Disconnected is displayed if no perimeter server is installed.

Exchange Process IDs

If either Connect:Direct or Sterling Integrator cannot exchange process identifiers, the process correlation in the various monitor user interfaces does not work for the particular instance of a process. When this occurs, correct the identifiers and restart the process.

Business Process Permissions

If you receive permissions-related error messages, make sure that adequate permissions have been assigned.

When a external request is received by Sterling Integrator to execute a business process, the user/proxy user must have business process permissions for the specific business process that is to be executed. That is, the business process itself must have permissions enabled when it is created. In addition, the Sterling user account must have:

- General business process execution permission. This is acquired by assigning the Business Processes group to the user account.
- Permission for the specific business process to be executed. The business process will display in the drop down with the mailboxes in the user account configuration windows.

Using the Graphical Process Modeler

The following restrictions apply to using the Graphical Process Modeler:

- Do not attempt simultaneous COPY operations for the same node.
- Issue a Close Session for each open session when session is finished.

Logging Files

More information, including audit messages and error messages, can be found in Sterling Integrator log files. Log files are available through the **Operations** > **System** > **Logs** menu.

The cdinterop.log files record details for all adapter activity and business processes that involve the Connect:Direct Server and Requester adapters and the related Connect:Direct services.

A new cdinterop.log is create each time the Sterling Integrator server is started. At midnight, the current log file is closed and a new one is created. In addition, when the size of the current log file reaches the maximum size specified in the cdinterop log file configuration screen, it is closed and a new log file is opened. The log file name includes a date and time stamp to make each name unique.

By default, the cdinterop.log file captures audit level messages, even when logging is not enabled. When the log is enabled, all transactions are recorded, including:

- Error messages
- Adapter startup and shutdown
- Adapter changes
- Security checks

Customizing the Connect:Direct Server Adapter Log Settings

You can enhance the logging for the Connect:Direct Server adapter using property files to reduce the footprint of the log files and processing overhead.

The log.properties file describes sets of properties required to define a logger used to log information to a file, including the cdinterop.log. This log.properties file should not be edited. If you need to override log.properties settings for the cdinterop.log, use the customer_overrides.properties file.

Note: Overriding the log.properties settings for the cdinterop.log can have a dramatic impact on performance and log volume.

The customer_overrides.properties file is used to override property settings in other property files. Unlike the other property files and their associated .properties.in files, the customer_overrides.properties file is not changed during installation of Application upgrades or patches. To prevent having your customized settings overwritten, you should use the customer override property file whenever possible rather than editing the Application property files or .in files.

The customer_overrides.properties file is not part of the initial Application installation and must be created. It must be named **customer_overrides.properties**.

To override log.properties file settings for the cdinterop.log:

1. In the `install_dir`/properties directory, locate (or create, if necessary) the customer_overrides.properties file.
2. Open the customer_overrides.properties file in a text editor.
3. Add the cdinteropcdinterop.log properties that you want to override to the log.properties file, using the following format:

   logService.`PROPERTY_NAME=PROPERTY_VALUE`

   logService-Name used to reference the log.properties file

   PROPERTY_NAME-Name of the property as used in the log.properties file

   PROPERTY_VALUE-The value you want to assign to the property

   For example, assume that you want to change the maximum number of cdinterop.log files to 5. To do so, override the cdinteroplogger.maxnumlogs property value in the log.properties file by adding the following line to the customer_overrides.properties file: `logService.cdinteroplogger.maxnumlogs=5`
4. Save and close the customer_overrides.properties file.
5. Stop and restart Application to use the new values.
6. Test your changes to ensure that the overrides give the desired results. If you have problems, contact Sterling Commerce Customer Support for assistance.

Configuration Settings

The following table describes properties used to configure the customer_overrides.properties file for the CDInterop logger:

Property	Description
logService.cdinteroplogger.logfilename	Specific name of the log file. Example: `ApplicationinstallDir/logs/cdinterop.log`

Property	Description
logService.cdinteroplogger.rotatelogs	Flag indicating whether to rotate the log after it has reached its maximum size. Example: true
logService.cdinteroplogger.maxlogsize	Maximum size of logs of this type. Example: 100000
logService.cdinteroplogger.maxnumlogs	Maximum number of logs of this type. Example: 10
logService.cdinteroplogger.loglevel	A level value of the logger. Valid entries: • NONE-Log nothing • CRITICAL-Log critical errors only • ERROR-Log errors only • WARN-Log errors and warnings • INFO-Log INFO and more severe • TIMING-Log errors, warnings, timing messages
logService.cdinteroplogger.displayname	Display name for the logger. Example: Log.CDInterop
logService.cdinteroplogger.showsource	Flag indicating whether to show the java class that originated an error message. **Note:** This can have a dramatic impact on performance and log volume, so it should only be used to diagnose problems. Example: false

Turning on Logging

If the error is not in the existing logs, turn on the Sterling Integrator cdinterop.log to capture activity of the Connect:Direct Server adapter and the services:

1. Browse to **Operation** > **System** > **Logs**.
2. Scroll to the Environment section of the page.
3. Click the edit icon to the left of Connect:Direct Server and Requester Adapter and Services.
4. In the Environment window, select On next to Logging Level and click Save.
5. Attempt to recreate the problem.
6. View the cdinterop log for more informative entries.

FIFO Enhancement

FIFO Message Processing

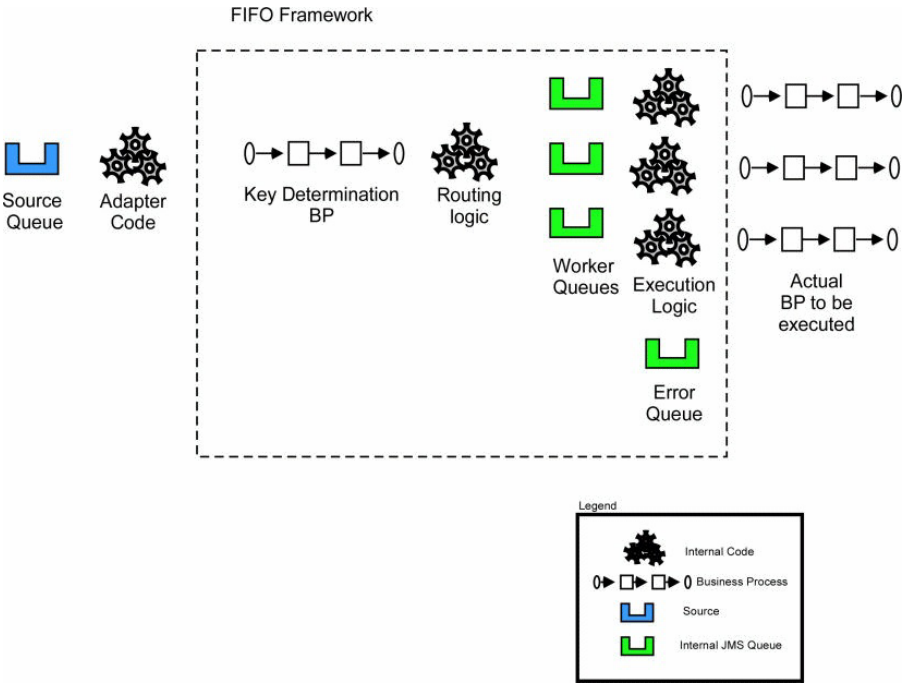
Sterling Integrator supports ordered processing of files and messages for the following adapters:

- JMS Queue adapter
- JMS Topic adapter

- MSMQ adapter

The ordered processing in Sterling Integrator is processed by the FIFO (first in first out) framework.

The following figure demonstrates the FIFO framework:



Sterling Integrator supports FIFO processing of messages through adapters. The messages passed to the FIFO framework are first executed through a specialized routing key initialization business process that returns a single string value known as the routing key. The routing logic is then applied, which places all the messages with equal keys on the same internal routing queue. Messages with different routing key values process in parallel. Messages with the same routing key value maintain FIFO ordering.

Each queue to user specified business process processes the message and waits for the business process to end the metadata describing the errant process, then processes the next message. If an error is encountered while processing the messages, metadata describing the errant process are routed to an error queue. Thereafter, the message processing continues.

Configuring FIFO Execution

You can customize the name and number of queues used in the FIFO framework. The number of task queues determines the number of concurrent processes that can execute in the system at a time. You can increase the number of queues, but it will consume more resources.

The queue is defined in the fifo.properties property file in the properties directory. All settings in the fifo.properties configuration file can be overridden via customer_overrides.properties. See the fifo.properties file for additional information pertaining to customer overrides.

The default queue configuration is as follows:

- `workflow.taskqueue.2=FIFO.GIS.QUEUE.2`
- `workflow.taskqueue.3=FIFO.GIS.QUEUE.3`
- `workflow.taskqueue.4=FIFO.GIS.QUEUE.4`
- `workflow.taskqueue.5=FIFO.GIS.QUEUE.5`
- `workflow.taskqueue.6=FIFO.GIS.QUEUE.6`
- `workflow.taskqueue.7=FIFO.GIS.QUEUE.7`
- `workflow.taskqueue.8=FIFO.GIS.QUEUE.8`
- `workflow.taskqueue.9=FIFO.GIS.QUEUE.9`
- `workflow.taskqueue.10=FIFO.GIS.QUEUE.10`
- `fifo.workflow.errorqueue=FIFO.GIS.ERROR`

FIFO Error Elements

The FifoError Type indicates the type of FIFO task that is being executed. At present, Async WorkFlow is the only type supported.

The table below lists the other FifoError elements:

Type	Description
TaskId	A unique ID given to each FIFO task executed by the FIFO framework.
TaskQueueId	The queue where the FIFO task was executed.
TaskQueueKey	The key that was returned through the FIFO routing key business process execution.
ErrorMessage	This element contains the information that assists in determining the cause of the failure.

WorkFlow Error Element

The table below lists the WorkFlow Error elements:

Type	Description
WorkFlowId	This element contains the workflow id that was executed.
WorkFlowContextId	This element contains the workflow context id for the first step of the business process. This information is used to retrieve the workflow and extract additional data in advanced scenarios.
WorkFlowInitiator	This element contains the name of the workflow initiator. In most cases, name of the adapter that started the process will be the workflow initiator name.
PrimaryDocumentId	This element contains the ID for the primary document of the business process.

FifoInitialization BPReport

This element contains metadata that describes the execution of the routing key initialization business process.

This is an optional node. It will be included both in process data of the executed business process and in the error queue XML. It is automatically included in the XML data if an error occurs during task initialization. To force the inclusion of this data, both in the error report and process data of the executed business process, ForceFifoInitializationDump to "true" in the routing key business process.

The table below lists the initialization BP report elements:

Type	Description
AdvancedStatus	This element contains the advanced status for the final step of this business process.
BasicStatus	This element contains the basic status for the final step of this business process.
PrimaryDocumentId	This element contains the primary document id at the last step of this business process.
ServiceName	This element contains the service name for the last step of this business process.
wfdName	This element contains the workflow definition name for this business process.
wfdVersion	This element contains the workflow definition version for this business process.
WorkFlowContextId	This element contains the workflow context id for this business process.
WorkFlowID	This element contains the workflow id for this business process.
StatusReport	This element contains the status report, if any, at the last step of this business process.
ProcessData	This element contains the process data at the last step of the business process.

FifoErrorNode Element

When the routing key business process is executed, the business process author can optionally write additional metadata to the FifoErrorNode element in the process data. This element and all the child nodes will be included in the FifoError document as part of this element.

The routing key business process has access to all process data information passed onto it through the adapter. See the example below for additional information about generating an error node.

```
<process name="AssignQueueKey">

  <sequence>

 <assign to="FifoRoutingKey"    from="DocToDOM(PrimaryDocument)/Order/@OrderId" />

   <assign to="FifoErrorNode/MSMQ/@QueueName" from="string(MSMQ/@QueueName)"  append="true"/>

  </sequence>

</process>
```

The additional information from the adapter can be included in the element to preserve the context of the error information in an easily identifiable manner.

FIFO Error Queue Listener

An out of the box adapter is configured on each node to listen to the error queue. This adapter is named "FIFO Error Queue Listener {nodename}". The adapter will bootstrap a business process named FifoError. This process is configured to retrieve the data from the errant process, including the original document and to integrate it into this process. This allows you to automate the re-processing of the data and other activities.

The FifoError process is defined as follows:

```
<process name="FifoError">

 <sequence>

    <operation>

     <participant name="FIFORouting" />

       <output message="Xout">

          <assign to="." from="*"></assign>

         <assign to="FifoTask">FifoErrorRecord</assign>

       </output>

        <input message="Xin">

         <assign to="." from="*"></assign>

       </input>

    </operation>

 </sequence>

</process>
```

The FifoError process provides a basic implementation for error handing. A user-specified business process may be configured to allow for customized error handling. A user-specified business process must contain the FIFORouting service as configured in the default FifoError process.

Recovering Errant Data

The messages in the error queue are written in XML format. The XML format provides information to determine the nature and source of the document containing the error. The error message contains information that enables the retrieval of document data; however, contents of the document are not stored in the message.

The error message format is as below:

```
<?xml version="1.0" encoding="UTF-8"?>

<FifoError ErrorMessage="" ErrorType="" TaskId="" TaskQueueId="" TaskQueueKey="" Type="">

  <WorkFlowError PrimaryDocumentId="" WorkFlowContextId="" WorkFlowId="" WorkFlowInitiator="">

   <FifoErrorNode/>

    <FifoInitializationBpReport AdvancedStatus="" BasicStatus="" PrimaryDocumentId="" ServiceName=""
 WfdName="" WfdVersion=""

      WorkFlowContextId="" WorkFlowId="">
```

```
        <StatusReport></StatusReport>
       <ProcessData>
          <PrimaryDocument SCIObjectID=""/>
        </ProcessData>
   </FifoInitializationBpReport>
  </WorkFlowError>
</FifoError>
```

FIFORouting Service

The FIFORouting service provides a control and reporting mechanism for interaction between business processes and the FIFO subsystem.

The FifoTask parameter specifies the task that this service should execute. Currently, there are two operational tasks this service provides: FifoResponse and FifoErrorRecord.

The FifoErrorRecord parameter specifies that the FIFORouting service should parse an error record from the error queue, retrieve the errant business process data, and report on it, as described above. This parameter should be used in conjunction with a retrieval of an error record from the error queue. The primary document in this mode of operation must be an FifoError XML record.

When executed in the FifoErrorRecord mode, the FIFORouting service will retrieve data pertaining to the errant business process and include it in ProcessData for the current business process. All data, including documents, may then be used directly within the current business process.

The service will generate data of the following format:

```
<ProcessData>
 ...
 <PrimaryDocument SCIObjectID=""/>
...
 <FifoProcess ErrorType="" WorkFlowContextId="" WorkFlowId=""
     WorkFlowInitiator="">
    <ProcessData>
     <FifoDetails>
        <FifoInitializationBpReport AdvancedStatus="" BasicStatus=""
           PrimaryDocumentId="" ServiceName="" WfdName="" WfdVersion=""
           WorkFlowContextId="" WorkFlowId="">
         <StatusReport>
          </StatusReport>
         <ProcessData>
            <PrimaryDocument SCIObjectID="" />
          </ProcessData>
       </FifoInitializationBpReport>
```

```
    </FifoDetails>
    </ProcessData>
 </FifoProcess>
</ProcessData>
```

Note: The first instance of ProcessData is that of the current error handler business process. The FifoProcess element contains the data from the errant business process. The ProcessData element within this element contains the data from the original errant business process. All data and documents within this ProcessData element may be used directly within this business process for error handing purposes.

The FifoReponse parameter specifies that the FIFORouting service should return a positive or negative success response to the FIFO subsystem. An optional parameter, FifoStatus, may also be specified. This status indicates whether or not the business process was a success and if it is an error, designations the FIFO subsystem to report an error. The FifoStatus parameter considers ERROR to be a failure and any other string data to be success.

The FifoResponse parameter is used to provide early response at to the success or failure of a FIFO business process. For example, assume business process A is the process that must be executed in FIFO. Business process A contains 10 steps. The first 5 steps must be executed in order; however, the last 5 steps provide data execution functionality where order is not important. In this example, optimal performance will be achieved by utilizing the FIFORouting service in FifoResponse mode to return the response at step 6. This will allow the next message to be processed immediately following the execution of this service and allow steps 7 through 11 to execute fully parallel.

Business Process Error Queue

The business process error queue is defined within the fifo.properties file. The error queue configuration defines the destination of errors within the FIFO framework. The error queue name should not contain spaces or punctuation.

The default business process error queue is shown below:

```
fifo.workflow.errorqueue=FIFO.GIS.ERROR
```

Business Process Queues

The FIFO business process execution queues are defined by rows that are prefixed with workflow.taskqueue. A queue row consists of a unique ID with prefix workflow.taskqueue to the left and a unique name without spaces or punctuation to the right.

You can add a queue by adding an additional row to the existing property file or to customer_overrides.properties. The simplest way to add additional queues is to continue the existing numbering scheme. You can remove a queue by deleting a row.

Note: Queues cannot be reduced below their default set of ten queues using customer_overrides.properties. If this is required, the queues must be removed directly from fifo.properties.

FIFO processing must be complete and the queues must be empty to change the queue configuration. You must disable the inbound adapter while changing the queue configuration. If the inbound adapter is not disabled and the queues are not drained, it may result in message execution that is out of order.

Cluster Configuration

The FIFO messaging system requires an external clustered JMS provider to allow proper execution and failover in a clustered configuration. An out of the box configuration for ActiveMQ 5.2 is provided to streamline this deployment. To configure FIFO messaging in a cluster for ActiveMQ:

1. Download ActiveMQ 5.2 from http://activemq.apache.org/activemq-520-release.html for the appropriate OS.
2. Deploy an instance of ActiveMQ 5.2 on each node of the cluster.
3. An activemq.xml file is included the properties/fifo directory of the Sterling Integrator deployment of each node. For each node, take this file and copy it to the ActiveMQ deployment on that node within the "conf" directory. This file will configure ActiveMQ to use failover clustering utilizing the Sterling Integrator database for storage and configure its port usage. By default, ActiveMQ will be configured to listen at the Sterling Integrator base port + 65 and the ActiveMQ interface will be at base port + 66 (http://server:base port + 66/admin).
4. On each Sterling Integrator node, the queue configuration must be re-directed to utilize the ActiveMQ cluster. In each node, add the following to customer_overrides.properties:

   ```
   fifo.broker.username=fifo.broker.password=fifo.broker.url=failover:
   (tcp://node1_hostname:node1_base_port + 65,tcp://
   node2_hostname:node_2_base_port + 65, ..., tcp://
   noden_hostname:node_n_base_port + 65)
   ```
5. Start the ActiveMQ instances on each node. See http://activemq.org for additional information about running an ActiveMQ instance.
6. Restart Sterling Integrator.

Configure FIFO Services

To configure FIFO services:

1. Login to Sterling Integrator.
2. Select **Deployment** > **Services** > **Configuration**.
3. Create new service and click **Go**.
4. In the Service Type field, enter the applicable adapter you want to use and click **Next**. You can also select it from the Tree View or List View.
5. Enter a suitable name and description in **Name** and **Description** fields.
6. Select or create a new group if required. By default, it is None.
7. Select the business process you want to execute. This business process must be set to use at least Minimal Event Processing and cannot be set to Error Only persistence level.
8. Select **FIFO** from Processing Mode drop-down list and click **Next**.
9. Select the business process that will receive the message and returns the routing key from the **FIFO Route Lookup BP** drop-down list. You should create a business process and import it into Sterling Integrator.
10. Review and click Finish.

The example below demonstrates routing key business process, which executes a set of XML documents in FIFO order by OrderID field:

```
<process name="AssignQueueKey">

  <sequence>

   <assign to="FifoRoutingKey"    from="DocToDOM(PrimaryDocument)/Order/@OrderId" />

  </sequence>

</process>
```

The routing information is not limited to XML documents only. Translation, Document Extraction, and other data extraction services can also be employed to retrieve routing data. In addition to the routing information in the document, the routing key business process has access to all information passed from the adapter in process data. If the routing key process fails, the error information will be placed in the *Business Process Error Queue*.

The routing key process must be configured with the *Enable Async Start Mode* disabled via the routing business process manager. If this is not configured, the routing key process will fail and the error information will be placed in the error queue.

Note: The FIFO Routing adapter must be enabled for message processing to occur. If this adapter is not enabled, messages will remain on the internal FIFO routing queues and no processing will occur.

ActiveMQ Data Storage

JDBC (Database) Master Slave is the default configuration for data storage employed to store FIFO data for ActiveMQ. In this configuration, each ActiveMQ node in a cluster is configured to utilize a single, shared database.

By default, this option is configured to make use of the existing Sterling Integrator database. As a result, this configuration option is setup out of the box and provides the simplest storage solution.

Shared File System Master Slave is an alternative data storage mechanism supported for FIFO, where a shared file system is used to store FIFO data for ActiveMQ. The shared file system option may yield better performance than when using JDBC.

Configure Shared File System Master Slave for ActiveMQ

You must manually configure the Shared File System Master Slave if you are not using the JDBC (Database) Master Slave configuration option to store FIFO data for ActiveMQ.

Note: Configuring FIFO messaging in a cluster for ActiveMQ is a prerequisite to configure the Shared File System Master Slave for ActiveMQ. For information on configuring FIFO messaging in a cluster, see *Cluster Configuration*.

To configure shared file system master slave for ActiveMQ:

1. In the activemq.xml file, comment out the following section:

   XML comments consist of the symbols, '<!--' to open the comment and '-->' to close the comment.

```
<!-- Database Storage Option -->

<!-- This section has been commented.

<persistenceAdapter>
      <jdbcPersistenceAdapter dataSource="#fifo-ds" useDatabaseLock="true">
        <statements>
          <statements tablePrefix="FIFO_"/>
        </statements>
      </jdbcPersistenceAdapter>
</persistenceAdapter>

-->
```

2. Uncomment the following section by removing the symbols '<!--' and '-->'.

```
<!-- File system Storage Option -->

 <persistenceAdapter>
    <journaledJDBC dataDirectory="/sharedFileSystem/broker"/>
</persistenceAdapter>
```

3. Edit the dataDirectory parameter to point to the location of the shared data directory to be used. This data directory must point to the same physical data location for all ActiveMQ instances in the network. For information on warnings about shared file system choices as a result of locking limitations, see *Shared File System Assumptions and Limitations*.
4. Restart each ActiveMQ node when you reconfigure it.

Shared File System Assumptions and Limitations

The following are some of the assumptions and limitations you must be aware of when using the Shared File System option to store FIFO data for ActiveMQ.

- Encrypted passwords for database storage are not currently supported. The file system based storage option described in this topic provides an alternative that does not require you to expose the database passwords.
- If an ActiveMQ node loses its connection to its database or file system storage, ActiveMQ will shut down. This is the intended behavior. Sterling Integrator currently does not employ out of the box monitoring for the ActiveMQ instances utilized for FIFO. To ensure seamless FIFO processing, the ActiveMQ nodes must be monitored and restarted if the instances are shut down for any reason.
- When ActiveMQ loses its database connection in conjunction with a Microsoft SQLServer database, ActiveMQ may hang during the shut down process. As a result, it may be difficult to determine if the ActiveMQ node has failed and requires to be restarted. It is recommended that you use the shared file system storage when using ActiveMQ in combination with a SQL Server database to avoid processing interruptions in failure scenarios.
- If you are reconfiguring any ActiveMQ options, ensure that you have executed all FIFO business processes. Failure to execute all FIFO business processes may result in the existing FIFO business processes remaining in an ‘Active’ state, in turn resulting in loss of FIFO ordering for the processes in the ‘Active’ state. To continue successful processing, the business processes in the ‘Active’ state will have to be manually halted and restarted.

nCipher and SafeNet/Eracom Support Enhancement

Using nCipher and SafeNet/Eracom Network and PCI Devices

Sterling Integrator supports the following nCipher and Safenet/Eracom devices:

Manufacturer	Device Types Supported
nCipher	Supports: • nShield series of PCI cards • NetHSM network devices
Safenet/Eracom	Supports: • ProtectServer Gold PCI card • ProtectServer Gold External network device • ProtectServer Orange PCI card • ProtectServer Orange External network device

Configure your Hardware Security Module (HSM)

Install and configure cards or HSMs according to the vendor's instructions. Ensure that java runtime components are available to interact with the device.

Sterling Integrator Features for HSM Support

An entry is stored in the CERTS_AND_PRI_KEY table by Sterling Integrator for each key pair and certificate. This entry contains information about:

- Keys and certificates, including the validity period, serial number, usage restrictions, issuer and subject used by the UI to display to the user without having to actually access the key or certificate.
- Normalizations of the distinguished name used by the system in searches.
- Modifications to the record.
- Certificate revocation status information.
- Keystore type.
- References to a binary keystore object stored in the DATA_TABLE. When a software keystore is used, the referenced object may contain key material. In the case of an HSM, it contains either reference information (nCipher) or a placeholder (Eracom).

OCSP Enhancement

Online Certificate Status Protocol (OCSP) Support in Sterling Integrator

Here's a little section in a concept.

The Online Certificate Status Protocol (OCSP) is a set of ASN.1 defined data structures for requesting and receiving information about certificate revocation status. These data structures can be sent and received by many transport protocols in principle. In practice, HTTP is used.

An OCSP client sends questions and processes responses. An OCSP responder answers questions and generates responses.

OCSP Client Functionality

An OCSP client implementation consists of the following:

- Data structures for managing information about OCSP responders
- Functionality for generating OCSP requests
- Functionality for processing OCSP responses
- Functionality for transmitting OCSP requests and receiving OCSP responses

How Sterling Integrator Performs an OCSP Check

About this task

An OCSP check for a certificate in Sterling Integrator is determined when the OCSP check within Sterling Integrator is implemented as a part of internal system APIs used by services for getting certificates and keys from the database. OCSP checks are performed by Sterling Integrator when methods are called to get certificates and keys from the objects that encapsulate them in the database.

The following steps describe how the OSCSP check is implemented in Sterling Integrator:

Procedure

1. The system examines the object that encapsulates the certificate to determine if OCSP checking is enabled. This allows the system to decide with no additional database calls whether to attempt an OCSP check.
2. If OCSP checking is enabled, the system gets the encoded issuer name from a certificate.
3. The system hashes the encoded issuer name with SHA1.
4. The system attempts to find an authority configured in the system that has a name whose hash matches that of the certificate.
5. If no authority is found, no check is performed.
6. If an authority is found, the system checks the OCSP policy for the authority. If the policy permits or requires OCSP checks, see the CERT_AUTHORITY table for more information. The system attempts to find an OCSP responder for the authority.
7. If no OCSP responder is found for the authority, one of the following happens:
   - If the authority policy is set to always check, an exception is thrown and the check fails.
   - If the authority policy is to only check when a responder is configured, no check is performed.
   - If an OCSP responder is found for the authority, an OCSP check is attempted.

Database Tables

Two new database tables have been added to manage OCSP-related information:

- CERT_AUTHORITY
- OCSP_RESPONDER

CERT_AUTHORITY

The CERT_AUTHORITY table maintains information about certificate authorities.

Column	Type	Description
OBJECT_ID	VARCHAR (255)	This is a GUID that constitutes a unique ID for a record. This is the primary key. Cannot be null.
NAME	VARCHAR (255)	A name for a record. Null allowed.
CREATE_DATE	DATETIME	A create date for a record.
MODIFIED_DATE	DATETIME	The date a record was last modified.
MODIFIED_BY	VARCHAR(255)	Information about who modified a record.
ISSUER_NAME	BLOB	The RDN of the authority taken from its certificate.
HASH_ALG	VARCHAR(128)	The hash algorithm used to compute name and key hashes. Only SHA1 is supported.
RDN_HASH	VARCHAR(255)	BASE64 encoded SHA1 hash of the DER encoded issuer RDN taken from the authority's certificate. This column is indexed.
KEY_HASH	VARCHAR(255)	BASE64 encoded SHA1 hash of the encoded public key in the issuer's certificate
CERT_OID	VARCHAR(255)	The OBJECT_ID of the authority's certificate in the CA_CERT_INFO table. Each authority must have a CA certificate in the database. Nulls not allowed.

OCSP_POLICY	VARCHAR(128)	The OCSP policy for the authority. This consists of two comma separated values. The values describe when to use OCSP and what to check. Possible values are: **OCSP_When** • never – never use OCSP • resp – use OCSP only if a responder is configured when a request is made • always – always use OCSP when a request is made. This requires a responder to be configured and will cause certificate checking to fail if no responder is configured **OCSP_What** • none – never check any certificates • end-user- Check only end user certificates • both – check both end-user and intermediate certificates. Currently not supported • Null is not allowed in this column
CRL_POLICY	VARCHAR(128)	Currently not used.
LOCK_ID	INTEGER	Used by the system to lock rows in the table.
CREATETS	TIMESTAMP	The timestamp of record creation for a row in the table.
MODIFYTS	TIMESTAMP	The last modification time for a row in the table.
CREATEUSERID	VARCHAR(40)	The user ID that created a row in the table.
MODIFYUSERID	VARCHAR(40)	The user ID that modified a row in the table.
CREATEPROGID	VARCHAR(40)	The name of a program or object that created a row in the table.
MODIFYPROGID	VARCHAR(40)	The name of a program or object that modified a record in the table.

OCSP_RESPONDER

The OCSP_RESPONDER table maintains information about OCSP responders.

Column	Type	Description
OBJECT_ID	VARCHAR (255)	This is a GUID that constitutes a unique ID for a record. This is the primary key. Cannot be null.
NAME	VARCHAR (255)	A name for a record. Null allowed.
CREATE_DATE	DATETIME	A create date for a record.
MODIFIED_DATE	DATETIME	The date a record was last modified.
MODIFIED_BY	VARCHAR(255)	Information about who modified a record.
ISSUER_NAME	BLOB	The RDN of the authority taken from its certificate.
HASH_ALG	VARCHAR(128)	The hash algorithm used to compute name and key hashes. Only SHA1 is supported.
RDN_HASH	VARCHAR(255)	BASE64 encoded SHA1 hash of the DER encoded issuer RDN taken from the authority's certificate. This column is indexed.
KEY_HASH	VARCHAR(255)	BASE64 encoded SHA1 hash of the encoded public key in the issuer's certificate
CERT_OID	VARCHAR(255)	The OBJECT_ID of the authority's certificate in the CA_CERT_INFO table. Each authority must have a CA certificate in the database. Nulls not allowed.
CACHE_TTL	VARCHAR(64)	The time in seconds to allow OCSP responses to live in the internal response cache If the column is NULL, OCSP responses will only be cached for 1 second, which in practice means not at all.
TRANS_PROF_OID	VARCHAR(255)	OBJECT_ID of a profile in the GIS database. You have to create a profile for the OCSP responder that includes the correct URL for the responder.

COMM_BP	VARCHAR(255)	Name of a business process to use to communicate with the OCSP responder. This has to be a business process that does HTTP communication. Services in the business process have to be configured to not require or present HTTP headers when sending and receiving, respectively. The process HTTPClientSend that comes with the system can be used and is recommended
COMM_WAIT	VARCHAR(24)	The number of seconds to wait for communication with the OCSP responder to take place before inferring that something is wrong.
LOCK_ID	INTEGER	Used by the system to lock rows in the table.
CREATETS	TIMESTAMP	The timestamp of record creation for a row in the table.
MODIFYTS	TIMESTAMP	The last modification time for a row in the table.
CREATEUSERID	VARCHAR(40)	The user ID that created a row in the table.
MODIFYUSERID	VARCHAR(40)	The user ID that modified a row in the table.
CREATEPROGID	VARCHAR(40)	The name of a program or object that created a row in the table.
MODIFYPROGID	VARCHAR(40)	The name of a program or object that modified a record in the table.
SEND_NONCE	VARCHAR(8)	Indicates whether to send a nonce with OCSP requests. Valid values: • true • false
REQ_NONCE	VARCHAR(8)	Indicates whether to require a nonce in OCSP responses. The system only recognizes the requirement for nonces on responses if it is required to send them in requests (SEND_NONCE=true). Valid values: • true • false

RESP_CERT_IN_CA_STORE	VARCHAR(8)	Indicates whether the certificate used to verify signatures on OCSP responses is in the CA store. Valid values: • true • false - The trusted store is checked.
RESP_CERT_OID	VARCHAR(255)	The object ID of the certificate used to verify signatures on OCSP responses. This is the object ID of a record in the CA_CERT_INFO or TRUSTED_CERT_INFO table.

OCSP Configuration

About this task

When configuring the system, you can create as many authorities and responders as you like.

To configure the system to use OCSP:

Procedure

1. Check the certificate for the certificate authority who issues the certificates you want to check in with OCSP into Sterling Integrator to verify it is a CA certificate.
2. List the CA certificates in the system and get the object ID for the certificate you just installed.
3. If the authority's OCSP response signing certificate is different than the authority's certificate issuing certificate, check the authority's OCSP response signing certificate into Sterling Integrator as a Trusted certificate.
4. If you checked in an additional OCSP signing certificate, list the Trusted certificates in the system and get the object ID for the certificate you just installed.
5. Go to the bin directory of the Sterling Integrator installation.
6. Start the database if necessary.
7. Start the bash or sh shell.
8. Source the file tmp.sh
9. Create an authority using the utility in the class com.sterlingcommerce.security.ocsp.SCICertAuthority.
10. Create an OCSP responder using the utility in the class com.sterlingcommerce.security.ocsp.SCIOCSPResponder
11. Update the certificates for the authority or individual certificates to enable OCSP. The utility com.sterlingcommerce.security.ocsp.SetAuthorityCertificatesOCSPInfo will configure all trusted and system certificates for an authority. The utility com.sterlingcommerce.security.ocsp.SetSystemCertificateOCSPInfo will

configure 1 system certificate. The utility com.sterlingcommerce.security.ocsp.SetTrustedCertificateOCSPInfo will configure 1 trusted certificate.

OCSP Configuration Scripts

The following scripts have been included with the OCSP hotfix to run the OCSP configuration utilities. There is a Unix/Linux and Windows version of each script. The scripts take the same command-line arguments as the utility programs they invoke. The scripts are located in the bin directory of the product install. The information about the command-line arguments is essentially just repeated in this section describing the scripts.

ManageCertAuthority.sh and ManageCertAuthority.cmd

Argument	Description
-a, -l, -d, -u2	Operation to perform: • -a - Add • -l - List • -d - Delete • -u2 - Update existing database record with newly computed key and RDN hashes. The –l option takes no additional arguments. The –d option takes a single argument: the object ID of the record to delete.
Name	Name of the authority. Required with -a.
Modified_by	User who modified or created the identity. Required with –a.
Hash_alg	Hash algorithm for the authority. Only the value “SHA1” is supported. Required with –a.
Certificate_id	Object ID of the CA certificate associated with the authority. Required with –a.

OCSP_policy	The OCSP policy string for the authority. This is a comma-delimited string as described in the section on the CERT_AUTHORITY table. Required with –a. For the first element of the string, the following are permitted: • never - Never use OCSP • resp - Use OCSP only if a responder is configured when a request is made • always - Always use OCSP when a request is made. This requires a responder to be configured and will cause certificate checking to fail if no responder is configured For the second element of the string, the following are permitted: **OCSP What** • none - Never check any certificates • end-user - Check only end user certificates • both - Check both end-user and intermediate certificates. Currently not supported. Examples: • never,none • always,end-user
Crl_policy	CRL policy string for the authority. Required with –a. A value is required for this argument, but it is not currently used. “None” is acceptable.
Object_ID	An object ID to use when creating this record. Optional with -a. Required with -u2.

ManageOCSPResponder.sh and ManageOCSPResponder.cmd

Argument	Description
-l	Gets a list of the currently configured OCSP Responders. This option takes no additional arguments.
-d	Deletes the configured OCSP Responder with the provided object ID for responders configuration data. This option takes object_id as an additional argument.

-u2	Updates existing records in the database with the correct information about the public key of the authority certificate and the subject DN of the authority certificate. This needs to be run against all existing records for both Cert Authority and OCSP Responders, or you need to delete and recreate the records to get the proper information into the database. This option takes object_id as an additional argument.
-a	Adds configuration data for a new OCSP Responder to be used for checking the status of certificates issued by the provided authority. Additional arguments are name, modified_by, hash_alg, authority_cert_oid, response_signing_cert_oid, resp_signing_cert_in_ca_store, cache_ttl, trans_prof_oid, comm_bp, comm_wait, send_nonce, require_nonce, and object_id.
name	(Required with -a) Name of the authority.
modified_by	(Required with -a) User who modified or created the identity.
hash_alg	(Required with -a) Hash algorithm for the authority. Only the value “SHA1” is supported.
authority_cert_oid	(Required with -a) Object ID of the CA certificate associated with the authority.
response_signing_cert_oid	(Required with -a) Object ID of the certificate that the provider of the OCSP services used to sign the response providing the status for the certificates. This certificate must be added to the CA Digital Certificate store or the Trusted Digital Certificate store. This is the System Certificate ID for the certificate as it appears in the store.
resp_signing_cert_in_ca_store	(Required with -a) Flag indicating if the previous value for the response_signing_cert_oid argument is found in the CA Digital Certificate Store in Sterling B2B Integrator.
cache_ttl	(Required with -a) The time-to-live in seconds for OCSP responses in the internal cache.
trans_prof_oid	(Required with -a) The object ID of a transport configured for communicating with the OCSP responder.

comm_bp	(Required with -a) Name of a business process to use to communicate with the OCSP responder. This has to be a business process that does HTTP communication. Services in the business process have to be configured to not require or present HTTP headers when sending and receiving, respectively. The process HTTPClientSend that comes with the system can be used and is recommended.
comm_wait	(Required with -a) The number of seconds to wait for communication with the responder until inferring that an error has occurred.
send_nonce	(Required with -a) Indicates if a NONCE value will be sent to the OCSP service. The NONCE value is used to prevent replay attacks by some OCSP providers.
require_nonce	(Required with -a) Indicates if the server should require that the OCSP service provide a NONCE value in the response.
object_id	(Optional with -a) An object ID to use when creating this record.

SetSystemCertOCSPInfo.sh SetSystemCerOCSPInfo.cmd

This utility will set the OCSP information in the database for a single system certificate

Argument	Description
-o, -n	How to interpret the second argument: -o object_ID -n name
Object_ID/Name	Object ID or name of the authority as determined by argument 1.

SetSystemCertOCSPInfo.sh and SetTrustedCertOCSPInfo.cmd

This utility will set the OCSP information in the database for a single system certificate

Argument	Description
-o, -n	How to interpret the second argument: -o object_ID -n name
Object_ID/Name	Object ID or name of the authority as determined by argument 1.

Run an OCSP Script

About this task

Use the following example to learn how to run the OCSP configuration scripts. These scripts assume that you have already checked in the CA certificates for the

authority, started the database, are in the bin directory of your Sterling Integrator install and have sourced the file tmp.sh in the bin directory.

After getting the object ID of the CA certificate from the authority, in Sterling Integrator from the Administration menu, select Trading Partners > Digital Certificates-CA. Select a certificate. The Certificate Summary dialog box appears with the certificate information, including its object ID.

Complete the following steps to run an OCSP Script:

Procedure

1. Run a command similar to the following to create an authority in the system:

   ```
   ./ManageCertAuthority.sh -a VPCA admin SHA1 "sedna:a1807c:11dc6d53ba4:-7b4b"
   "always,end-user" "none"
   ```

2. After creating an authority, and creating a profile for communicating with an OCSP responder, run a command similar to the following to create an OCSP responder in the system:

   ```
   ./ManageOCSPResponder.sh -a VPCA admin SHA1 "sedna:a1807c:11dc6d53ba4:-7b4b"
    "2400" "a1807c:11dc79aacbd:-7570" HTTPClientSend 3600
   ```

3. Run a command similar to the following to list all of the authorities in the system:

   ```
   ./ManageCertAuthority.sh -l
   ```

   Return output for each authority displays:

   ```
   CERT_AUTHORITY:
   OBJECT_ID: sedna:1ded0fd:11dc9d22929:-7fbd
   NAME: VPCA
   CREATE_DATE: 2008-11-23
   MODIFIED_DATE: 2008-11-23
   MODIFIED_BY: null
   ISSUER_NAME: Country=US, StateOrProvince=Dublin,
   OrganizationUnit=GIS Development,Organization=Sterling,
   CommonName=Test CA
   HASH_ALG: SHA1
   RDN_HASH: 24E63F8AE9F51497529EA0CC34467A4680737A9F
   ENCODED_RDN_HASH: JOY/iun1FJdSnqDMNEZ6RoBzep8=
   KEY_HASH: C96F2FF442EBFA07672DCEC49B729D4D24898313
   ENCODED_KEY_HASH: yW8v9ELr+gdnLc7Em3KdTSSJgxM=
   CERT_OID: sedna:a1807c:11dc6d53ba4:-7b4b
   OCSP_WHEN_POLICY: always
   OCSP_WHAT_POLICY: end-user
   CRL_POLICY: null
   ```

4. Use a command similar to the following to enable OCSP for all trusted and system certificates issued by the authority:

   ```
   ./SetAuthorityCertsOCSPInfo.sh -o "sedna:1ded0fd:11dc9d22929:-7fbd" yes
   ```

OCSP Check Logic

About this task

The following steps describe the logic of OCSP checking in Sterling Integrator:

If the certificate status is ok, the OCSP check succeeds. Otherwise, it fails.

Procedure

1. If an existing response whose time-to-live has not expired is found, than that response is used as the OCSP response.

2. If no existing response is found in the cache or the time-to-live has expired for a response in the cache, an OCSP request is created.
3. If the system creates an OCSP request, it launches the business process configured for the OCSP responder to send the request and get the response. Requests will include a nonce value if the responder was configured to have one sent.
4. If the business process completes successfully, the system attempts to parse its primary document as an OCSP response. The business process used to send OCSP requests and receive OCSP responses strips the HTTP headers from the response.
5. If the primary document can be parsed as an OCSP response, the system checks the status of the response.
6. If the response status indicates that the request generated a valid response, the system attempts to verify the signature on the OCSP response using the certificate configured for the OCSP responder.
7. If the signature is verified and the responder was configured to require nonce, the system attempts to get and check the nonce from the response.
8. If all other verifications passed, then the system looks for certificate status information for the certificate for which the request was constructed and sent.
9. If the status information is found, then the system updates the internal cache for an existing OCSP response for the certificate.

Producer/Consumer Relationship Report Enhancement

The producer/consumer relationship reports are used to view the mailbox producer and consumer relationships. This report provides information on the:

- Producer Partner Name
- Producer Mailbox
- Consumer Mailbox
- Policy Settings
- Routing Rules

The following table lists the available producer/consumer relationship reports:

Report Name	Description
ConsumerProducerRelationships	Organized by consumer name. All other available criteria is reported according to the defaults.
ProducerConsumerRelationships	Organized by producer name. All other available criteria is reported according to the defaults.

Use **Operations** > **Reports** to run this report.

Shared and Linked Mailboxes Enhancement

The Shared mailbox functionality allows you to instantly share real-time data with the trading partners. You can use the Linked mailbox functionality to link individual trading partner's mailboxes with one or more shared mailboxes. Linking trading partner mailboxes to shared mailboxes allows the trading partners to view the real-time data stored in the shared mailboxes. In other words, a linked mailbox

provides a link to view the data in a shared mailbox. The linked mailbox is a read-only copy of the shared mailbox and the data in the mailbox cannot be modified or deleted.

To enable the shared and linked mailbox functionality, set the mailbox.enableSharedLinkedMailboxes property to true in the customer_overrides.properties file. By default, this property is set to false.

A linked mailbox can be related to only one shared mailbox. Multiple linked mailboxes can point to the same shared mailbox. Adding data to a shared mailbox immediately makes that data available to all links.

A Regular or a Shared mailbox can be the parent of a Shared mailbox. A Regular or a Linked mailbox can be the parent of a Linked mailbox. A sub-mailbox under a Shared or Linked mailbox must be the same mailbox type as its parent.

A user's regular mailbox can contain a combination of regular, linked, and shared mailboxes. Shared mailboxes and linked mailboxes need not be in the same directory. User permissions can be explicitly applied to linked mailboxes. Routing rules cannot be applied to linked mailboxes. However, virtual roots can be applied directly to linked mailboxes.

The following are some of the limitations when converting a regular mailbox to a shared or linked mailbox and converting a shared or linked mailbox to a regular mailbox:

- A regular mailbox can be converted to a shared or linked mailbox type.
- A shared mailbox can be converted to a regular mailbox type only if the parent of the shared mailbox is a regular mailbox type. The links to the shared mailbox should be removed before converting the shared mailbox to a regular mailbox.
- A linked mailbox can be converted to a regular mailbox type only if the parent of the linked mailbox is a regular mailbox type.
- A shared mailbox cannot be converted to a linked mailbox.
- A linked mailbox cannot be converted to a shared mailbox.
- A regular mailbox with sub-mailboxes can be converted to a linked mailbox type when all the sub-mailboxes are of the linked mailbox type.
- A regular mailbox with sub-mailboxes can be converted to a shared mailbox type when all the sub-mailboxes are of the shared mailbox type.

View a List of Mailboxes

About this task

To view a list of mailboxes:

Procedure

1. From the **Deployment** menu, select **Mailboxes** > **Configuration.**
2. Open the configuration data of the mailbox you want to view using one of the following methods:
   - In the **By Mailbox Name** field of the **Search** section, type the name or partial name of the mailbox you want to view and click **Go!**
   - In the **Alphabetical** section, select the letter the mailbox starts with or select all to pull a list of all mailboxes and click **Go!**

3. A list of available mailboxes opens. The following table describes the content of each column:

Column Title	Description
Select	Contains the icons for • editing a mailbox • deleting a mailbox • creating a sub-mailbox
Mailbox Name	Displays the name and path of the mailbox
Description	Displays the short description of the mailbox
Type	Displays the type of Mailbox: • R - Regular - Standard mailbox that cannot be linked to any other mailbox • S - Shared - Mailbox that can be linked from other mailboxes • L - Linked - Mailbox whose data is stored in a shared mailbox
Linked To	Displays the shared mailbox path where linked mailboxes are pointing to. **Note:** You must remove links to the shared mailboxes before deleting a shared mailbox.
Last Modified	Displays the timestamp to indicate when the mailbox was last modified.

Create a Shared Mailbox

About this task

To create a shared mailbox:

Procedure

1. From the **Deployment** menu, select **Mailboxes** > **Configuration.**
2. In the **Create** section, click **Go!**
3. On the **Mailbox: Name** page, select the parent mailbox in which the mailbox you are creating will be embedded. You can type a partial name in the **Filter by Name** field and click the **filter** button for a filtered list. The root mailbox is denoted by a slash (/).
4. In the **Name** field, type a name for the mailbox you want to create. This name is used to identify the mailbox in the application.
5. In the **Description** field, type a short description for the mailbox. Use this field to describe the mailbox. This is a required field. This field is not used by any other resource in the system.
6. In the **Mailbox Type** field, select `Shared` as the type of the mailbox you want to create from the following options:
   - Regular (Default)
   - Shared
   - Linked

   When creating linked sub-mailboxes, the available shared sub-mailboxes will be restricted to those belonging to its parent's shared mailbox.
7. Click **Next**.

8. OIn the **Assign Groups** page, use the arrows to add the groups to the **Selected Groups** list and click **Next**. All groups in the **Selected Groups** list will have permissions on this mailbox. Click the first double arrow to add all available groups to the **Selected Groups** list. You can type a partial group name in the **Filter by Name** field and click the **filter** button for a filtered list. No groups are required. Groups can be added from the **Accounts** menu.
9. Use the arrows to add users to the **Selected Users** list and click **Next**. All users in the **Selected Users** list will have permissions on this mailbox. Click the double arrow to add all available users to the **Selected Users** list. You can type a partial user name in the **Filter by ID** field and click the **filter** button for a filtered list. No users are required. Users can be added from the **Accounts** menu.
10. On the **Confirm** page, verify your mailbox configuration and click **Finish**.

Create a Linked Mailbox

About this task

To create a linked mailbox:

Procedure

1. From the **Deployment** menu, select **Mailboxes** > **Configuration.**
2. In the **Create** section, click **Go!**
3. On the **Mailbox: Name** page, select the parent mailbox in which the mailbox you are creating will be embedded. You can type a partial name in the **Filter by Name** field and click the **filter** button for a filtered list. The root mailbox is denoted by a slash (/).
4. In the **Name** field, type a name for the mailbox you want to create. This name is used to identify the mailbox in the application.
5. In the **Description** field, type a short description for the mailbox. Use this field to describe the mailbox. This is a required field. This field is not used by any other resource in the system.
6. In the **Mailbox Type** field, select `Linked` as the type of the mailbox you want to create from the following options:
   - Regular (Default)
   - Shared
   - Linked

   When creating linked sub-mailboxes, the available shared sub-mailboxes will be restricted to those belonging to its parent's shared mailbox.
7. Click **Next**.
8. OIn the **Assign Groups** page, use the arrows to add the groups to the **Selected Groups** list and click **Next**. All groups in the **Selected Groups** list will have permissions on this mailbox. Click the first double arrow to add all available groups to the **Selected Groups** list. You can type a partial group name in the **Filter by Name** field and click the **filter** button for a filtered list. No groups are required. Groups can be added from the **Accounts** menu.
9. Use the arrows to add users to the **Selected Users** list and click **Next**. All users in the **Selected Users** list will have permissions on this mailbox. Click the double arrow to add all available users to the **Selected Users** list. You can type a partial user name in the **Filter by ID** field and click the **filter** button for a filtered list. No users are required. Users can be added from the **Accounts** menu.

10. On the **Confirm** page, verify your mailbox configuration and click **Finish**.

SFTP with Mailbox Login without Virtual Root Permission Enhancement

Using SFTP with Mailboxes

A *Mailbox* is a storage area for *messages*. Each message associates a name with some data (the data itself is stored in Sterling Integrator as a *document*.) Mailboxes are usually arranged in a hierarchy with the mailbox named "/" serving as the root.

Mailboxes in Sterling Integrator are analogous to the familiar directory structure offered by operating systems' file systems. A Mailbox is a directory and messages correspond to files in the directory.

Mailboxes are more feature rich than the normal file system. A mailbox can be configured to invoke a business process when a message is sent to it. Messages have well defined extractability policies that govern the conditions under which messages can be successfully extracted (retrieved).

The SFTP Server adapter uses system Mailboxes as the repository. The prerequisites to using SSH/SFTP are:

- One or more Mailboxes set up as the repository for SFTP
- Users with appropriate permissions to SFTP mailboxes
- Users configured with virtual root or with the *MailboxLoginWithoutVirtualRootPermission* permission

Using SCP with Mailboxes

A *Mailbox* is a storage area for *messages*. Each message associates a name with some data (the data itself is stored in Sterling Integrator as a *document*.) Mailboxes are usually arranged in a hierarchy with the mailbox named "/" serving as the root.

Mailboxes in Sterling Integrator are analogous to the familiar directory structure offered by operating system file systems. A Mailbox is a directory and messages correspond to files in the directory.

Mailboxes are more feature rich than the normal file system. A mailbox can be configured to invoke a business process when a message is sent to it. Messages have well defined extractability policies that govern the conditions under which messages can be successfully extracted (retrieved).

The SFTP Server adapter uses Sterling Integrator Mailboxes as the repository. The prerequisites to using SSH/SCP in Sterling Integrator are:

- One or more Mailboxes set up as the repository for SCP
- Users with appropriate permissions to SCP mailboxes
- Users configured with virtual root or with the *MailboxLoginWithoutVirtualRootPermission* permission

SFTP Mailboxes

The SFTP Server adapter uses Mailboxes as the repository. To use SSH/SFTP:

- Set up one or more Mailboxes as the repository for SFTP
- Assign users appropriate permissions to SFTP mailboxes

- Configure users with virtual root or with the *MailboxLoginWithoutVirtualRootPermission* permission

Chapter 3. Build 5101 or Higher

Mailbox Permissions Enhancement

The following table lists a set of permissions added in this release to enable a non-admin user to execute schedules and business processes in the Sterling Integrator Mailbox:

Permission Name	Description
UI BP Execution Administrator	This permission enables an operator (non-admin) to execute a business process manually by specifying a different user in the **Run as User** field in the BP Execution page.
UI Schedule Administrator	This permission allows a user to administer all schedules in the system regardless of which user is specified in the **Run As User** field. The user can create, search, edit, and execute the schedules.
UI Scheduler	This permission displays the scheduler menu in the user interface. The user can also create, search, edit, and execute the schedules that run as this user.
UI Schedule Reviewer	This permission allows a user to view all schedules and only modify the schedules that run as this user.

Notices

This information was developed for products and services offered in the U.S.A.

IBM may not offer the products, services, or features discussed in this document in other countries. Consult your local IBM representative for information on the products and services currently available in your area. Any reference to an IBM product, program, or service is not intended to state or imply that only that IBM product, program, or service may be used. Any functionally equivalent product, program, or service that does not infringe any IBM intellectual property right may be used instead. However, it is the user's responsibility to evaluate and verify the operation of any non-IBM product, program, or service.

IBM may have patents or pending patent applications covering subject matter described in this document. The furnishing of this document does not grant you any license to these patents. You can send license inquiries, in writing, to:

IBM Director of Licensing

IBM Corporation

North Castle Drive

Armonk, NY 10504-1785

U.S.A.

For license inquiries regarding double-byte character set (DBCS) information, contact the IBM Intellectual Property Department in your country or send inquiries, in writing, to:

Intellectual Property Licensing

Legal and Intellectual Property Law

IBM Japan Ltd.

19-21, Nihonbashi-Hakozakicho, Chuo-ku

Tokyo 103-8510, Japan

The following paragraph does not apply to the United Kingdom or any other country where such provisions are inconsistent with local law:
INTERNATIONAL BUSINESS MACHINES CORPORATION PROVIDES THIS PUBLICATION "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Some states do not allow disclaimer of express or implied warranties in certain transactions, therefore, this statement may not apply to you.

This information could include technical inaccuracies or typographical errors. Changes are periodically made to the information herein; these changes will be

incorporated in new editions of the publication. IBM may make improvements and/or changes in the product(s) and/or the program(s) described in this publication at any time without notice.

Any references in this information to non-IBM Web sites are provided for convenience only and do not in any manner serve as an endorsement of those Web sites. The materials at those Web sites are not part of the materials for this IBM product and use of those Web sites is at your own risk.

IBM may use or distribute any of the information you supply in any way it believes appropriate without incurring any obligation to you.

Licensees of this program who wish to have information about it for the purpose of enabling: (i) the exchange of information between independently created programs and other programs (including this one) and (ii) the mutual use of the information which has been exchanged, should contact:

IBM Corporation

J46A/G4

555 Bailey Avenue

San Jose, CA 95141-1003

U.S.A.

Such information may be available, subject to appropriate terms and conditions, including in some cases, payment of a fee.

The licensed program described in this document and all licensed material available for it are provided by IBM under terms of the IBM Customer Agreement, IBM International Program License Agreement or any equivalent agreement between us.

Any performance data contained herein was determined in a controlled environment. Therefore, the results obtained in other operating environments may vary significantly. Some measurements may have been made on development-level systems and there is no guarantee that these measurements will be the same on generally available systems. Furthermore, some measurements may have been estimated through extrapolation. Actual results may vary. Users of this document should verify the applicable data for their specific environment.

Information concerning non-IBM products was obtained from the suppliers of those products, their published announcements or other publicly available sources. IBM has not tested those products and cannot confirm the accuracy of performance, compatibility or any other claims related to non-IBM products. Questions on the capabilities of non-IBM products should be addressed to the suppliers of those products.

All statements regarding IBM's future direction or intent are subject to change or withdrawal without notice, and represent goals and objectives only.

All IBM prices shown are IBM's suggested retail prices, are current and are subject to change without notice. Dealer prices may vary.

This information is for planning purposes only. The information herein is subject to change before the products described become available.

This information contains examples of data and reports used in daily business operations. To illustrate them as completely as possible, the examples include the names of individuals, companies, brands, and products. All of these names are fictitious and any similarity to the names and addresses used by an actual business enterprise is entirely coincidental.

COPYRIGHT LICENSE:

This information contains sample application programs in source language, which illustrate programming techniques on various operating platforms. You may copy, modify, and distribute these sample programs in any form without payment to IBM, for the purposes of developing, using, marketing or distributing application programs conforming to the application programming interface for the operating platform for which the sample programs are written. These examples have not been thoroughly tested under all conditions. IBM, therefore, cannot guarantee or imply reliability, serviceability, or function of these programs. The sample programs are provided "AS IS", without warranty of any kind. IBM shall not be liable for any damages arising out of your use of the sample programs.

Each copy or any portion of these sample programs or any derivative work, must include a copyright notice as follows:

© IBM 2015. Portions of this code are derived from IBM Corp. Sample Programs. © Copyright IBM Corp. 2015.

If you are viewing this information softcopy, the photographs and color illustrations may not appear.

Trademarks

IBM, the IBM logo, and ibm.com® are trademarks or registered trademarks of International Business Machines Corp., registered in many jurisdictions worldwide. Other product and service names might be trademarks of IBM or other companies. A current list of IBM trademarks is available on the Web at “Copyright and trademark information” at http://www.ibm.com/legal/copytrade.shtml.

Adobe, the Adobe logo, PostScript, and the PostScript logo are either registered trademarks or trademarks of Adobe Systems Incorporated in the United States, and/or other countries.

IT Infrastructure Library is a registered trademark of the Central Computer and Telecommunications Agency which is now part of the Office of Government Commerce.

Intel, Intel logo, Intel Inside, Intel Inside logo, Intel Centrino, Intel Centrino logo, Celeron, Intel Xeon, Intel SpeedStep, Itanium, and Pentium are trademarks or registered trademarks of Intel Corporation or its subsidiaries in the United States and other countries.

Linux is a registered trademark of Linus Torvalds in the United States, other countries, or both.

Microsoft, Windows, Windows NT, and the Windows logo are trademarks of Microsoft Corporation in the United States, other countries, or both.

ITIL is a registered trademark, and a registered community trademark of the Office of Government Commerce, and is registered in the U.S. Patent and Trademark Office.

UNIX is a registered trademark of The Open Group in the United States and other countries.

Java™ and all Java-based trademarks and logos are trademarks or registered trademarks of Oracle and/or its affiliates.

Cell Broadband Engine is a trademark of Sony Computer Entertainment, Inc. in the United States, other countries, or both and is used under license therefrom.

Linear Tape-Open, LTO, the LTO Logo, Ultrium and the Ultrium Logo are trademarks of HP, IBM Corp. and Quantum in the U.S. and other countries.

Connect Control Center®, Connect:Direct®, Connect:Enterprise®, Gentran®, Gentran®:Basic®, Gentran:Control®, Gentran:Director®, Gentran:Plus®, Gentran:Realtime®, Gentran:Server®, Gentran:Viewpoint®, Sterling Commerce™, Sterling Information Broker®, and Sterling Integrator® are trademarks or registered trademarks of Sterling Commerce®, Inc., an IBM Company.

Other company, product, and service names may be trademarks or service marks of others.

Product Number:

Printed in USA